BUPT_MCPRL at TRECVID 2023: Video to Text Description and ActEV SRL Challenge

Yang Song , HongPu Zhang , ZeLiang Ma , Shuai Jiang , Zhe Cui, Yanyun Zhao
Beijing University of Posts and Telecommunications
{sy12138 , zhp , mzl , js , cuizhe , zyy}@bupt.edu.cn

Abstract

In this report, we present our solutions to the Video to Text (VTT) task and Activities in Extended Video (ActEV) task released in TRECVID 2023.

Firstly, we presented a solution for the Video to Text Description (VTT) task of TRECVID 2023[1]. We employed a large-scale vision-language model with an Encoder-Decoder architecture as the backbone, pre-trained and fine-tuned on the task-specific dataset[2]. Specifically, videos were encoded to features by BLIP2[3] and then adapted for VTT tasks through fine-tuning with the Q-Former mechanism. These features were then fed into the Large Language Model (LLM), leveraging a cross-attention mechanism to generate complete textual descriptions. Furthermore, to acquire high-quality captions, we introduced a cyclic data augmentation method wherein model-inferred captions were processed using a threshold-based matching approach for data expansion. To enhance the model's focus on semantic representation in videos, we innovatively explored the LLM's ability to integrate fine-grained semantic information from multiple captions.

Through the approach we proposed, significant improvements have been achieved in the overall performance of the system. The submitted results secured the second position in terms of CIDEr, CIDErD, and BLEU metrics, with a CIDEr score of 0.879, trailing the first place by only 0.03. Additionally, we attained the first position in the SPICE metric.

Moreover, our proposed 5 classification methods and learning strategy greatly benefit the performance, ranking first on TRECVID'23 ActEV Self-Reported Leaderboard (ActEV-SRL) Challenge on the MEVA dataset. The PMiss@0.1rfa indicator is as high as 0.5781, far exceeding the second-place result.

1 Video to Text (VTT)

1.1 Introduction

Video to Text Description (VTT) is a task that involves generating textual descriptions of scenes, events, and objects appearing in videos. It stands as one of the most challenging tasks in the field of computer vision, requiring models to handle data from two different modalities, namely videos and text, while also effectively modeling details, semantics, and temporal information.

Currently, the generation paradigm for video-to-text typically leverages the feature representations of powerful vision-language models. By utilizing a visual encoder, the video

stream is encoded to obtain high-dimensional vector embeddings, while on the text side, a text encoder is employed to acquire high-dimensional word embeddings. Using the foundational Transformer architecture and cross-attention mechanism, these two modalities' feature representations are interactively combined to generate word-based descriptions of the video content.

However, the current opening datasets for video-text caption are not as abundant as image-text datasets, and video-text caption is more complicated than image-text caption in terms of capturing more context-relevant video descriptions.

To address these issues, we firstly established overall system based on the BLIP2 model[3], called Bootstrapping Language-Video Pre-training(BLVP), which underwent pre-training for image-text tasks. Then we introduced a cyclic data augmentation approach for the scarcity of data in VTT tasks. Moreover, in response to the challenge of relatively weak semantic information extraction during the transfer process, the semantic analysis capability of the LLM was studied and applied to this task, which allowed us to integrate information from the video stream to produce comprehensive descriptions of the videos.

1.2 Related Works

VTT task is one of the most challenging research topics in the field of video understanding. Early researchers employed templated-based methods[4]. The S2VT approach introduced a sequence-to-sequence method based on LSTM for video-to-text generation[5].

In recent years, Vision-Language Pretraining(VLP) was successfully advanced the developments of feature representation and understanding, resulting in significant acceleration in downstream tasks like image captioning[6].

Recently, BLIP[7], BLIP2, and VALOR[8], which benefit from large-scale training data for visual encoders and text decoders, exhibited strong generalization capabilities. For example, in the TRECVID 2022 competition, the RUCAIM3-Tencent[9] utilized the Encoder-Decoder architecture of BLIP and emerged as the winners of the competition. Similarly, the LVC_HDU[10] achieved promising results by employing a large-scale visual encoder along with an LSTM-based video decoder for captioning tasks. Drawing on current visual-language approaches and the experience of previous VTT teams, we developed our solution for VTT, as detailed in Section 3.

1.3 Method

In this section, we introduce the overall workflow of our system. The overall system workflow is illustrated in Fig. 1, including model input (a), model training (b), caption filtering (c) and semantic description synthesis (d). In this Fig., (a), (b) and (c) constitute the cyclic data augmentation section, while (d) represents the multi-model fusion of LLM. Next, we focus on the model structure we used, the approach of cyclic data augmentation, and the utilization of LLM.

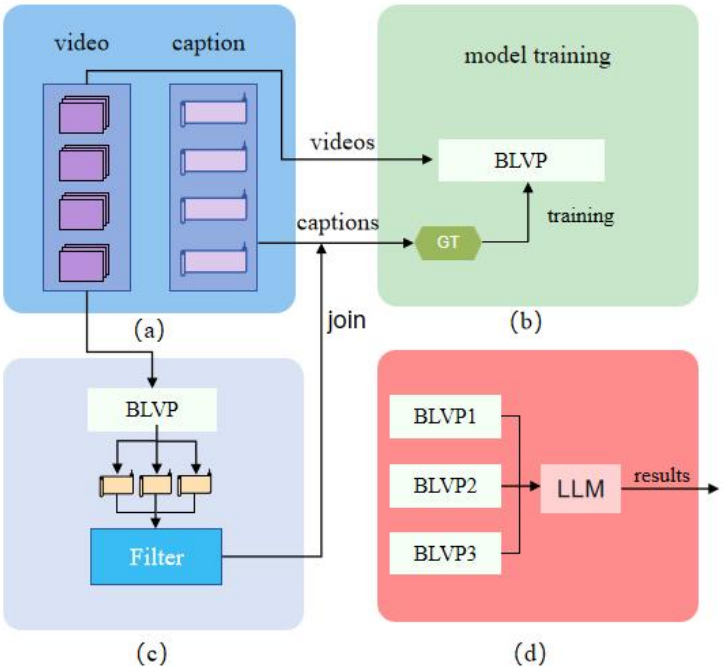

Fig. 1. The overall system workflow.

1.3.1 Model Structure

Given the strong performance of BLIP2 in addressing visual-text problems, this paper leverages this network architecture as a foundational structure and applies it to the VTT task, enabling it to adapt to the task of generating textual descriptions from video content. The image encoder in BLIP2 was transformed to video encoder.

BLVP that we propose is depicted in Fig. 2. In contrast to BLIP2, the input visual features consists of multiple frames rather than a single frame, while we maintained the structure of the image encoder in order to retain the pre-trained weights. Specifically, each video was sampled at 16 frames to constitute a video clip. The extracted features were fed into cross-attention with the Q-Former. Subsequently, the output was passed to the Open Pre-trained Transformers (OPT) model[11] , which generates textual descriptions used to convey the content of the videos.

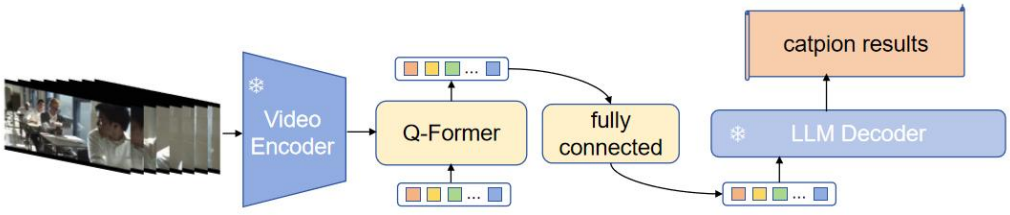

Fig. 2. The model structure of BLVP.

1.3.2 Data Augmentation

Obviously, the video-text caption model training requires a large enough training data set. However, it is worth noting that the existing video-text caption dataset repositories do not exhibit the same level of abundance as image-text counterparts. This is because that acquiring context-relevant video descriptions is notably intricate. In TRECVID 2023 VTT evaluation task, we are also limited by the problem of insufficient data set. To address this issue, we proposed a cyclic data augmentation solution as shown in Fig. 1. The training process and data augmentation process were conducted alternately.

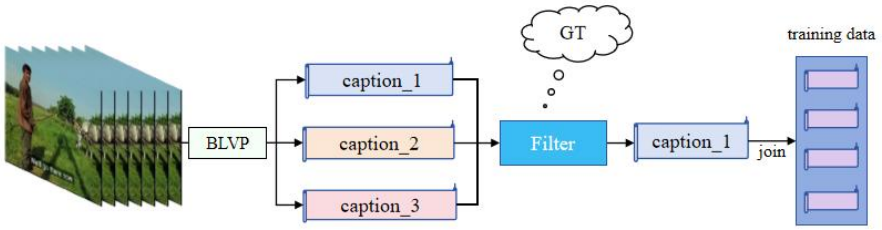

Fig. 3. The process of data selection.

The proposed cyclic data augmentation can be divided into two stages: training stage and data augmentation stage. During the training stage, the model is trained based on the weights retained from the previous training stage. In the data augmentation stage, we infer the training set with the model trained in last training stage and obtain the text description of videos. We use these text descriptions to calculate CIDEr scores against the ground truth, and then select highly-quality text description data to add to the raw training dataset for the subsequent training process. The selecting process of high-quality text descriptions is illustrated in Fig. 3, and Filter is responsible for selecting highly-quality text descriptions whose CIDEr score is greater than the threshold ($\mathrm{th}_{\mathrm{CIDEr}}$) we set, such as caption-1 in Fig. 3. Selected text results closely align with the raw dataset.

It is worth noting that, in early epoches of model training, the text description results of the i+1 round of inference tend to be of higher quality than the i round of inference. Therefore, in each training round, we utilize captions from the previous round along with the raw data, rather than accumulating captions from all rounds. It also helps save on training time and training resources. For specific experimental details and parameter settings, please refer to Section 4.

1.3.3 LLM for Ensemble

During the process of fine-tuning, we observed that the system performance could be improved by using multiple different texts generation models or increasing the size of model parameters. However, there exist limitations to the performance gains that can be made by increasing the size of the model parameters. We also found that different models can generate high-quality captions for intuitive and semantically clear videos, and they tend to perform poorly on complex, multi-object, and noisy videos. Therefore, how to accurately extract text descriptions of complex video containing multiple objects is an important factor affecting the overall system performance. Based on the above, we crafted three models to perform inference on the same video and simply concatenated their outputs (text descriptions), as shown in Fig. 4. Additionally, we

employed the open-source LLM model, Alpaca-Lora 7B[12] , to integrate these three sentences into a logically coherent single-sentence description. The prompt used for this integration is as follows:

"I will provide you with three sentences describing the same video, summarize their semantic information, and use the words found in these sentences to craft a single sentence description, maintaining the style of the original three sentences."

This approach allows us to extract semantic information from a textual perspective and integrate information effectively. It is particularly beneficial for videos with complex scenes and significant variations in descriptions, ultimately leading to an improvement of the overall performance of the system.

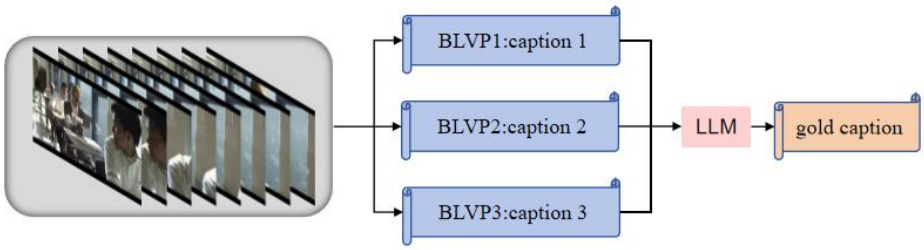

Fig. 4. The overview architecture of LLM for ensemble.

1.4 Experiments

1.4.1 Experiments Details

Our primary experiments were conducted based on BLIP2. We froze the image encoder and text decoder, fine-tuning on the Q-Former component. In the proposed cyclic data augmentation, the filter selected data with a CIDEr score greater than 0.7 for inclusion as new data. Each training stage comprises 5 epochs. And the equipment and parameters used during training are as follows: Our experiments were conducted on two NVIDIA RTX 4090 GPUs with a batch size of 6. Each frame was resized to 364x364 pixels, and the text decoder used the OPT2.7 version. We performed training for five epochs at a time, followed by cyclic data augmentation. The learning rate was set at 1e-5, and the training employed the prompt "a video of " . We used 32 queries for training.

1.4.2 Experiments results

The submitted results secured the second position in terms of CIDEr, CIDErD, and BLEU metrics, with a CIDEr score of 0.879, trailing the first place by only 0.03. Additionally, The experimental results of each team are shown in Fig. 5.

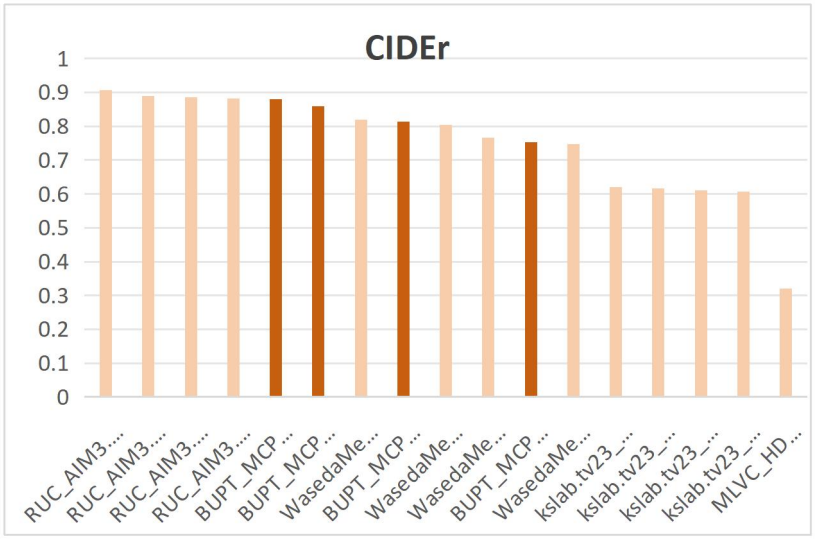

Fig. 5. Result comparison of all teams in TRECVID 2023 VTT task[1].

The results obtained through LLM ensemble in Sec 3.3 have attained the first position in the SPICE metric. The SPICE metrics of each team are illustrated in the Fig. 6.

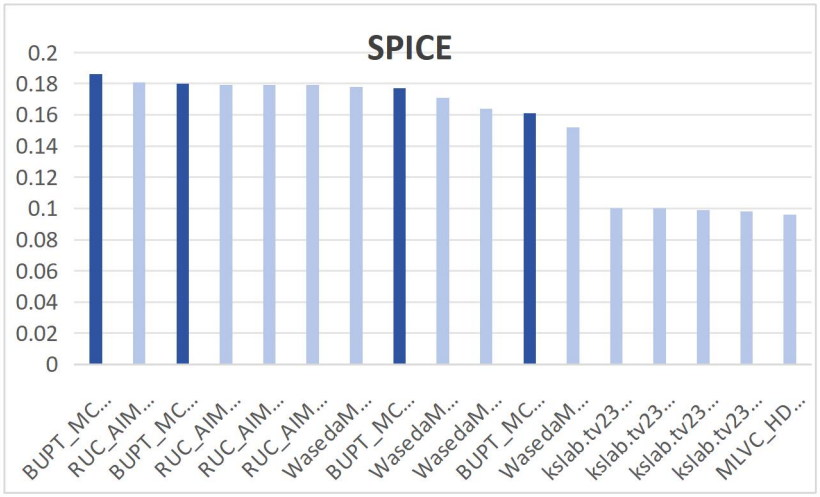

Fig. 6. The SPICE metrics of all teams in TRECVID 2023 VTT task[1].

1.4.3 ablation experiments

Fig. 7 and 8 illustrate the changes in data quantity achieved through data augmentation in Sec 3.2 , as well as the consistent upward trend of CIDEr score on the validation set. This experiment serves as evidence of the effectiveness of our cyclic data augmentation method in Sec 3.2.

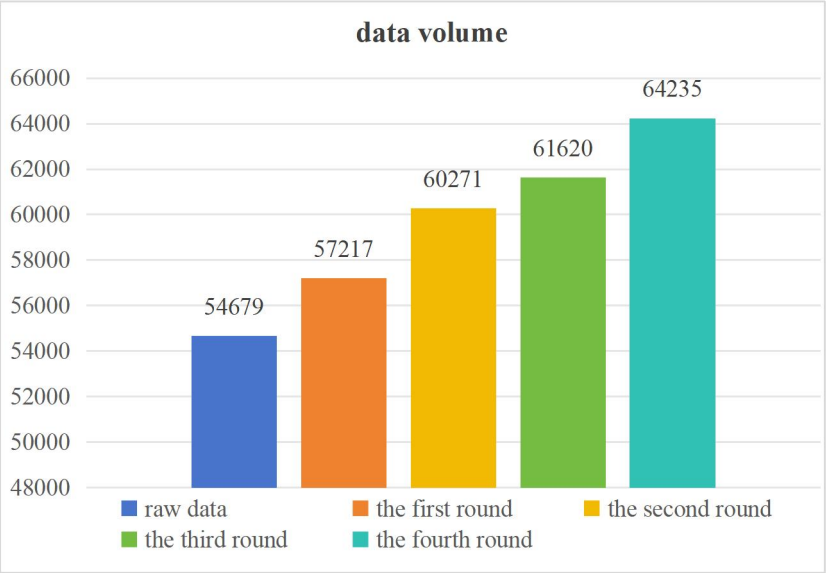

Fig. 7. Graph of data volume changes with increasing expansion rounds.

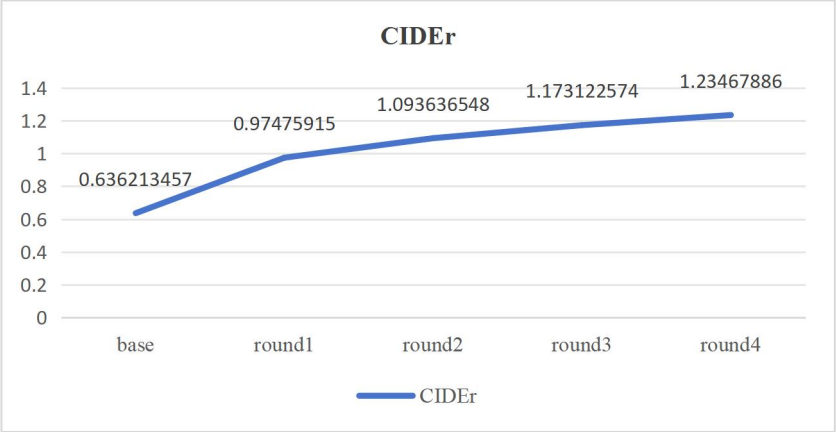

Fig. 8. Change in CIDEr Score with increasing expansion rounds.

1.5 Conclusion

In this study, we transferred the BLIP2 model to the VTT task and proposed a cyclic data augmentation approach. Our experimental results on the TRECVID VTT dataset achieved a CIDEr score of 87.9, ranking second in the competition. Additionally, to extract the semantic information, we used LLM to integrate the results of multiple models, obtaining a state-of-the-art SPICE evaluation metric. However, there is room for improvement in our temporal modeling approach, particularly in understanding complex motion behaviors. Additionally, due to time constraints, we were only able to perform cyclic data augmentation, for a limited five rounds.

2 Activities in Extended Video (ActEV)

2.1 Introduction

1. Training data: MEVA, COCO (pretraining), Kinetics400 (pretraining).

2. Approach: (27401: 2D+3D detectors, 5 different classification methods)

3. Difference: None.

4. Contribution: Our proposed 5 classification methods and learning strategy greatly benefit the performance, ranking first on TRECVID'23 ActEV Self-Reported Leaderboard (ActEV-SRL) Challenge on the MEVA dataset. The PMiss@0.1rfa indicator is as high as 0.5781, far exceeding the second-place result.

5. Conclusion: Surveillance video scene activity recognition is complex, for different activities, the classification methods should be specifically analyzed and designed.

2.2 Method

The MEVA dataset contains a total of 20 categories of activities, which we roughly divide into 5 activity groups and process them separately:

1) person-object: *person_reads_document, person_texts_on_phone, person_picks_up, person_puts_down, person_sits_own, person_stands_up, person_transfers_object;*

2) person-specific object: *person_interacts_with_laptop;*

3) vehicle-only: *vehicle_starts, vehicle_stops, vehicle_turns_left, vehicle_turns_right;*

4) person-vehicle: *person_exits_vehicle, person_enters_vehicle, person_opens_vehicle_door, person_closes_vehicle_door;*

5) scene-related and person-person: *person_opens_facility_door, person_enters_scene_through_structure, person_exits_scene_through_structure, person_talks_to_person.*

Notably, our grouping strategy differs from the one[13] that ranked first last year in the "***person-object***" and "***person-specific object***" categories. In the "***person-specific object***" category, only the "*person_interacts_with_laptop*" class is present.

Based on this, we propose a comprehensive surveillance video activity detection framework as shown in Figure 9. For each activity group, the video is split into **sequence**, sent to the 3D detector, and then connected into trajectories as **proposal clips**. These trajectories are input into a specially trained classifier to obtain the **classification scores** for each proposal clips. VideoMAEv2[14] has been selected as our classifier due to its outstanding performance in video feature representation and action recognition. Finally, all clips go through a post-processing module specially designed for class-specific characteristics to obtain the **Activity trajectories**. Concretely, apart from the ***person-person***, we develop different methods to detect and recognize the activities for the different activity groups respectively. For the ***person-person***, the approach outlined in [13] was retained, and various hyperparameters were fine-tuned. In this chapter, we will introduce the framework of our system separately for each category of activity groups.

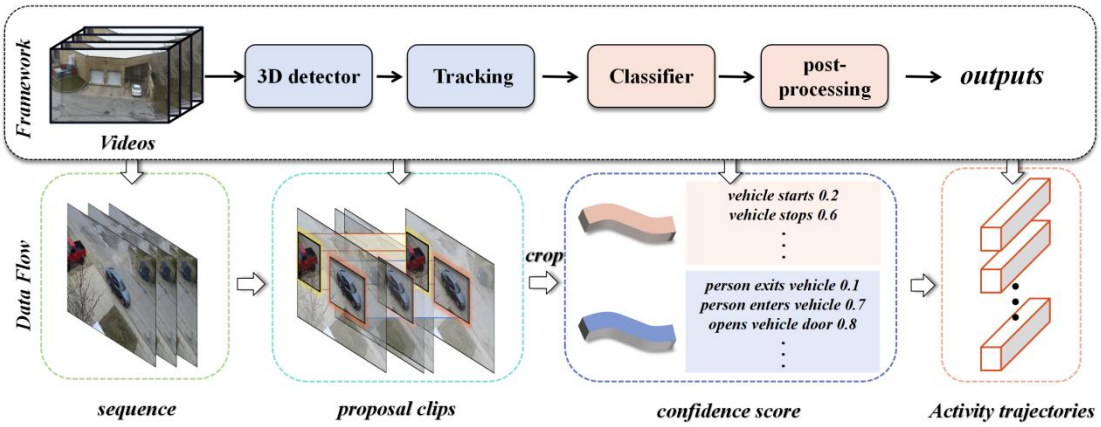

Figure 9. The framework of our activity detection.

2.2.1 person-object

For ***person-object*** activity group detection, as shown in Figure 10, we first detect the activity proposals in time-space domain with Cascade RCNN 3D[15] from the video clips, then link the activity proposals according to their IoUs to form an activity trajectory. After that, the activity trajectory is fed into the classifier trained for this activity group to be classified. Ultimately, we get the final activity tracks for each class with the t-NMS[16] post processing. The same as[13], we also adopt the classification score merge strategy, but the difference is that we add another classifier VideoMAEv2[14]. Specifically, we use Video Swin Transformer[17], ActionCLIP[18] and VideoMAEv2 as the classifiers, and merge the scores for each category by weighted average. And for post-processing, we use different parameters for each category to get better activity detection performance. In order to improve the performance of activity region detection, we combined six categories of activity samples on ***person-object*** group and one category on ***person-special object*** group to train activity proposal detector.

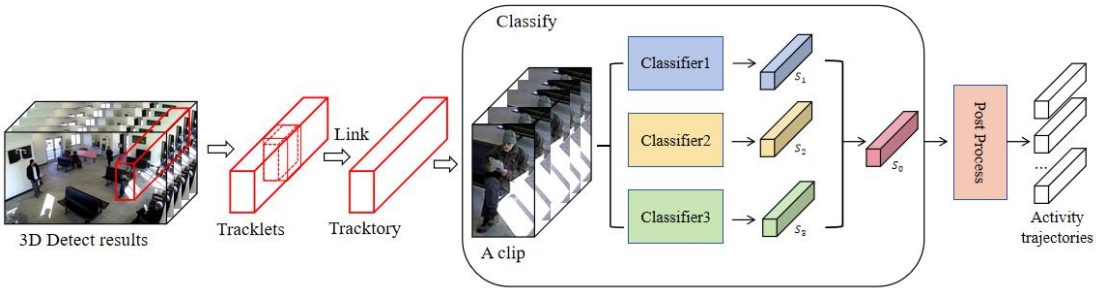

Figure 10. The overview of person-object activity detection

2.2.2 person-specific object

For ***person-specific object*** group, we use the same activity proposal detector and classifier as person-object group. The difference is that *person_interacts_with_laptop* has scenes on the test set that are not found in the training set and validation set, resulting in poor performance in this category, so we incorporated other detection results. Specifically, as shown in Figure 11, we straightly use YOLOv8[19] to detect the laptop that appears in each frame, and use KAPAO[20] to detect the bounding box of the person and the key points of the wrist joint. After obtaining the detection results, we use the Hungarian algorithm[21] to match the laptop and the corresponding

person, where the cost matrix is the distance from the center point of the laptop to the key point of the wrist. Finally, the maximum external rectangle of the person and laptop matched is used as the detection bounding box. After merging this result, the detection performance of *person_interacts_with_laptop* has been greatly improved.

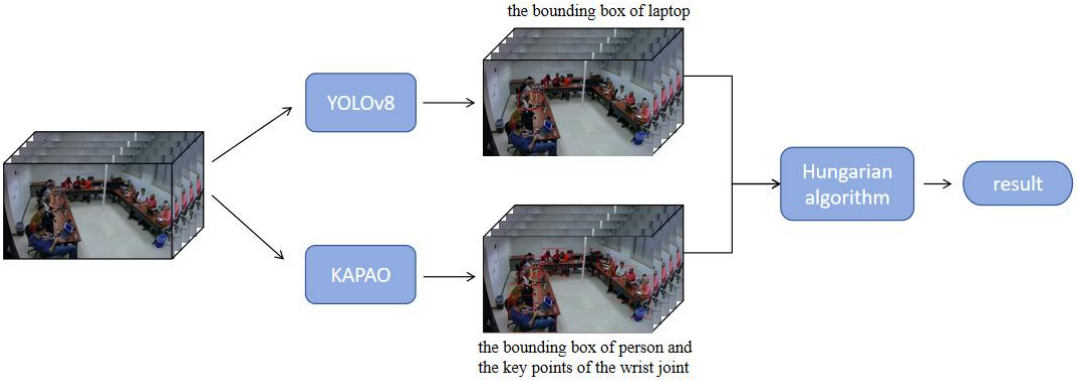

Figure 11. The overview of person-specific object activity detection

2.2.3 vehicle-only

In this activity group, we continue to use the framework illustrated in Figure 9 but have specially designed specialized modules based on category-specific characteristics. For the start-stop category, we set different anchor box lengths to detect the behavioral regions and employ velocity as an auxiliary feature for classification. For the left-right turning category, we improve the reverse classification model[13] to filter out the reversing cases so as to obtain accurate localization of activities in temporal domain as far as possible .

The four categories can be divided into two mutually exclusive groups: *vehicle_starts* and *vehicle_stops*, and *vehicle_turns_left* and *vehicle_turns_right*. To achieve higher performance, we fine-tuned VideoMAEv2 separately on these two groups. Additionally, we conducted a statistical analysis of the duration distribution in each category within the training set. As shown in Figure 12(a), there are distinct differences between the duration of vehicle start-stop and turn left-right behaviors. The duration distribution for start-stop behaviors resembles a nearly uniform distribution ranging from approximately 40 to 150 frames, while the duration distribution for left-right turns is similar to a *Poisson* distribution. Therefore, we employ varying time window lengths for the inference of different behaviors in 3D detection. For vehicle start-stop activity detection, the time window sizes of video clips are selected as (64, 96, 128), and for left-right turning activity detection, the time window sizes chosen are (96, 128, 160, 192). By setting time anchors of different sizes, the activity detector can capture clips containing complete behaviors, enabling the classifier to classify accurately.

Additionally, we visualized the trend of speed changes over time for the validation set samples. Figure 12(b) exhibits an ascending trend in the speed for *vehicle_starts* activities and a descending trend for *vehicle_stops* activities. Hence, based on the velocity correlation between the start-stop categories, we designed a post-processing module to assist in the classification of this two categories. Experiments have shown that these methods can enhance the performance of activity recognition.

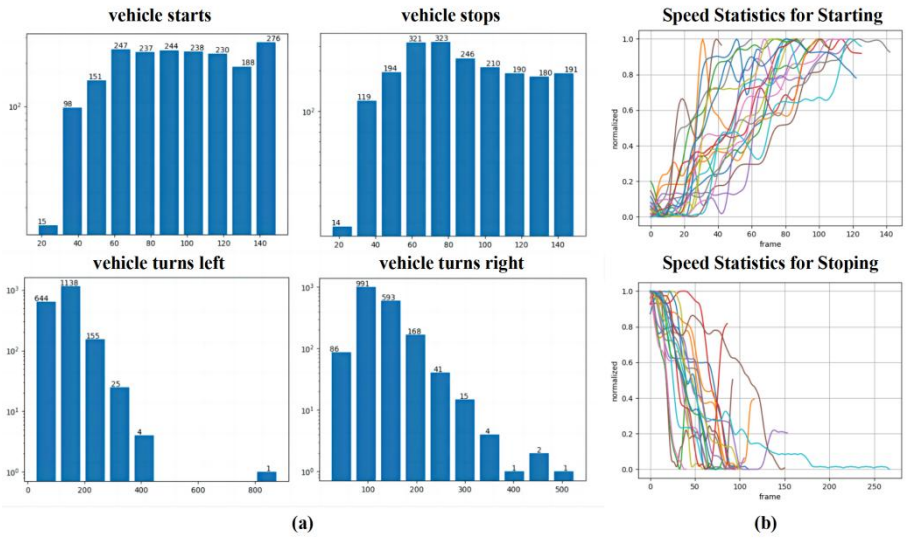

Figure 12. The statistics of activity duration for vehicle-only group and the visualization of moving speed for start-stop activities. (a) shows the statistical distribution of activity duration for vehicle-only categories, with the horizontal axis representing the number of frames of activity duration and the vertical axis representing the number of samples. (b) presents the velocity statistics for vehicle start-stop categories, with the horizontal axis representing the frame sequence of samples and the vertical axis representing the normalized speed.

Note that, the documentation[13] clearly indicates that the reversing category does not belong to the left-right turning category. Therefore, before feeding proposal clips into the left-right turning classifier, filtering of clips is necessary. Since the training dataset contains relatively few samples of the reversing category, directly fine-tuning the big model could lead to overfitting. To address this issuse, we take a subset of the left-right turning category samples and reverse them, subsequently adding the generated data to the reversing category dataset. This approach alleviates overfitting and significantly improves the reversing classifier's accuracy.

2.2.4 person-vehicle

For this group, we use the same framework as vehicle-only group as shown in Figure 9, where behavioral regions are initially detected, and clips for classification are subsequently generated through trajectory linking. In contrast, during the training phase of the classifier, we adopted a multi-task training approach. For the same Backbone, we designed two separate prediction heads, dedicated to predicting the categories of opening and closing vehicle doors and entering and exiting the vehicle, respectively. This approach not only ensured an ample number of samples in the dataset but also allowed the backbone to learn shared semantic features of both sub-groups. The prediction heads focused on capturing the distinctive features of their respective exclusive categories. Empirical evidence has demonstrated that this approach results in higher classification accuracy compared to fine-tuning two separate models.

However, owing to the Non-Maximum Suppression (NMS) operation applied to activity detection results, the situations involving multiple behaviors within a confined area are consolidated into a single behavioral region. For instance, when four people exit a car simultaneously, it is detected as one region, and only one result can be obtained when it is input into the classifier. Consequently, the cases where multiple activities occur within one region cannot be handled by this framework.

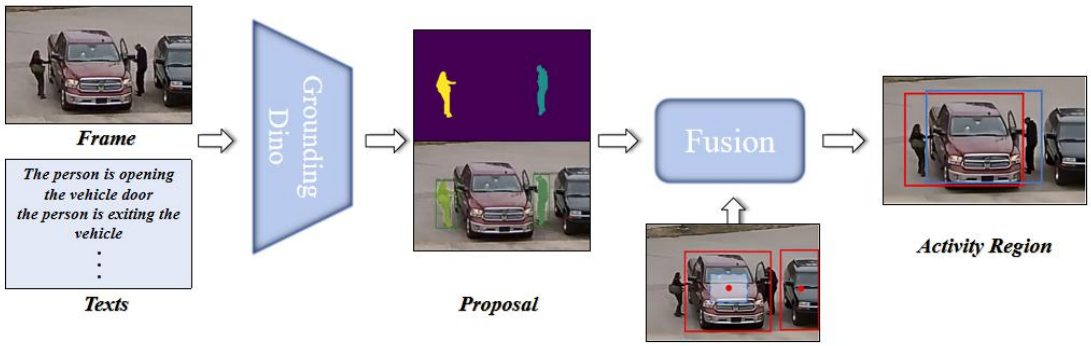

Figure 13. The overview of person-vehicle activity detection.

To address this issue, as illustrated in Figure 13, the Grounding DINO[22] model is employed to detect individuals engaged in specific behaviors, reducing the missed detections of multiple behaviors in the same region. Additionally, the location information of the vehicles is located to establish fine-grained behavioral regions, ensuring that only one behavior occurs within each designated region. With this approach, we effectively solve the detection problem where multiple people interact with the same vehicle to form multiple concurrent behaviors.

2.2.5 scene-related

In this activity group, the entering and exiting scenes often involve multiple instances of people interacting with each other. For example, after the first person opens a door, others follow and enter the scene one by one. This situation of multi-person following behavior may lead to the detection module to detect only one behavior area and miss the remaining multiple behaviors entering the scene. In other words, the 3D detection method may not be effective in such cases.

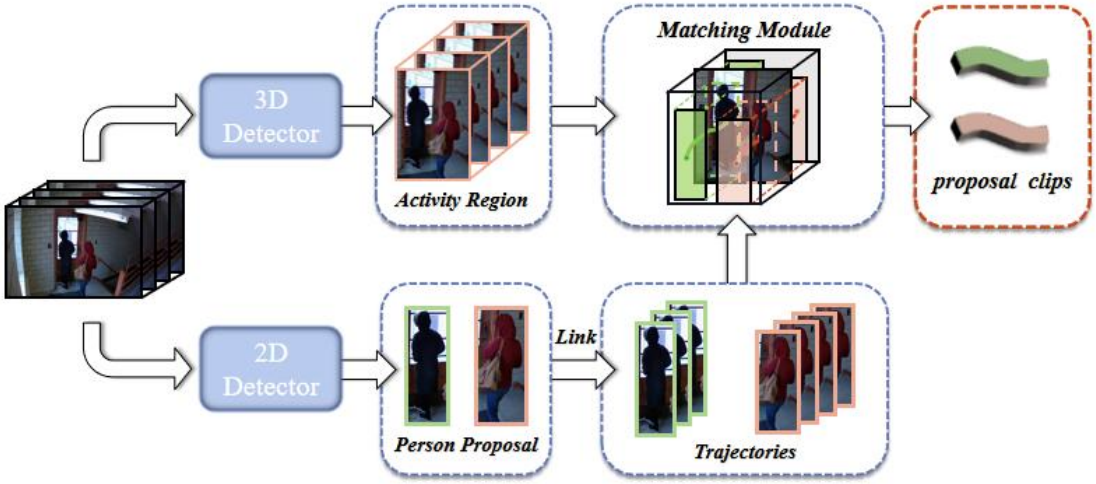

Figure 14. The overview of scene-related activity detection.

Therefore, we have also introduced a 2D detection approach. As illustrated in Figure 14, we employ the state-of-the-art YOLOv8 model to detect individuals within each scene and use the DeepSORT[23] algorithm to track and generate trajectories for each person. Subsequently, we calculate the Intersection over Union (IoU) between the trajectories and the behavioral regions detected by 3D detection. This matching process allows us to associate different individuals' trajectories with the behavioral regions in which actions are occurring. When the two sets of

results match, it indicates that different individuals are involved in actions within the same region. The results of this matching help in expanding the behavioral regions to finely differentiate areas where following behaviors are taking place.

2.3 Result

The ActEV-SRL challenge based on the MEVA dataset has strict requirements on the time and space of the system output, so we use different detectors and classifiers to meet the needs of different categories. Our system achieves PMiss@0.1rfa=0.5781 and won the first place on the MEVA dataset, which proves the effectiveness of our method.

Table 1. Results in TRECVID 2023 ActEV SRL Evaluation[24]

Team	Pmiss
BUPT_MCPRL	0.5781
mlvc_hdc	0.8952
WasedaMeiseiSoftbank	0.9985
FDU_AWS	0.9999
406	1
qwer1	1
hsmw	1

Reference

[1] http://www-nlpir.nist.gov/projects/tvpubs/tv23.papers/tv23overview.pdf

[2] Rossetto, Luca, et al. "V3C – a research video collection." MultiMedia Modeling: 25th International Conference, MMM 2019, Thessaloniki, Greece, January 8 – 11, 2019, Proceedings, Part I 25. Springer International Publishing, 2019.

[3] Li, J., Li, D., Savarese, S., & Hoi, S. (2023). "Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. " arXiv preprint arXiv:2301.12597.

[4] Guadarrama, Sergio, et al. "Youtube2text: Recognizing and describing arbitrary activities using semantic hierarchies and zero-shot recognition." Proceedings of the IEEE international conference on computer vision. 2013.

[5] Venugopalan, Subhashini, et al. "Sequence to sequence-video to text." Proceedings of the IEEE international conference on computer vision. 2015.

[6] Li, X., X. Yin, and Oscar LI C. "Object-Semantics Aligned Pre-training for Vision-Language Tasks [C]." European Conference on Computer Vision. Springer, Cham. 2020.

[7] Li, Junnan, et al. "Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation." International Conference on Machine Learning. PMLR, 2022.

[8] Chen, Sihan, et al. "Valor: Vision-audio-language omni-perception pretraining model and dataset." arXiv preprint arXiv:2304.08345 (2023).
[9] Yue, Zihao, et al. "RUCAIM3-Tencent at TRECVID 2022: Video to Text Description." Proceedings of TRECVID. 2022.
[10] Li, Ping, Tao Wang, and Xingchao Ye. "MLVC_HDU@ TRECVID 2022: Video to Text (VTT) and Activities in Extended Video (ActEV) Tasks."
[11] Zhang, Susan, et al. "Opt: Open pre-trained transformer language models." arXiv preprint arXiv:2205.01068 (2022).
[12] Andermatt, Pascal Severin, and Tobias Fankhauser. "UZH_Pandas at SimpleText@ CLEF-2023: Alpaca LoRA 7B and LENS Model Selection for Scientific Literature Simplification." (2023).
[13] Zhao H, Tong Z, Xiao Y, et al. BUPT-MCPRL at TRECVID 2022 ActEV SRL Challenge[J].
[14] Wang L, Huang B, Zhao Z, et al. VideoMAE V2: Scaling Video Masked Autoencoders with Dual Masking Supplementary Material[J].
[15] Cai Z, Vasconcelos N. Cascade r-cnn: Delving into high quality object detection[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 6154-6162.
[16] Neubeck A, Van Gool L. Efficient non-maximum suppression[C]//18th international conference on pattern recognition (ICPR'06). IEEE, 2006, 3: 850-855.
[17] Liu Z, Ning J, Cao Y, et al. Video swin transformer[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022: 3202-3211.
[18] Wang M, Xing J, Liu Y. Actionclip: A new paradigm for video action recognition[J]. arXiv preprint arXiv:2109.08472, 2021.
[19] https://github.com/ultralytics/ultralytics
[20] McNally, William, et al. "Rethinking keypoint represent 其 4ations: Modeling keypoints and poses as objects for multi-person human pose estimation." *European Conference on Computer Vision*. Cham: Springer Nature Switzerland, 2022.
[21] Japrapto B A .Hungarian Algorithm[J]. 2010.
[22] Liu S, Zeng Z, Ren T, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection[J]. ar**v preprint ar**v:2303.05499, 2023.
[23] Wojke N, Bewley A, Paulus D. Simple online and realtime tracking with a deep association metric[C]//2017 IEEE international conference on image processing (ICIP). IEEE, 2017: 3645-3649.
[24] https://actev.nist.gov/SRL#tab_leaderboard